

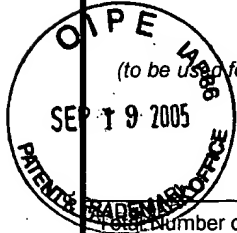
# 112 2700

PTO/SB/21 (08-00)

Approved for use through 10/31/02. OMB 0651-0031

Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paper Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.



# TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number

09/631,884

Filing Date

August 4, 2000

First Named Inventor

Jyotirmoy Paul

Group Art Unit

2176

Examiner Name

Nathan Hillery

Attorney Docket Number

50277-0352

Total Number of Pages in This Submission  
(including this Transmittal)

24

## ENCLOSURES (check all that apply)

- ☒ Fee for Appeal \$500.00
- ☒ Fee Enclosed
- ☐ Amendment / Response
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Response to Missing Parts/Incomplete Application
  - ☐ Response to Missing Parts under 37 CFR 1.52 or 1.53

- ☐ Assignment Papers (for an Application)
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition To Convert To a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, number of CD(s) \_\_\_\_\_

- ☐ After Allowance Communication to Group
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☐ Appeal Communication to Group (Appeal Notice, Brief, Reply Brief)
- ☐ Proprietary Information
- ☐ Status Letter
- ☒ Other Enclosure(s) (please identify below):

Appeal Brief (23 pgs); and  
Acknowledgment Receipt  
postcard

Remarks

## SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm  
or  
Individual name

Hickman Palermo Truong & Becker LLP  
Stoycho Draganoff, Reg. No. 56,181

Signature

Stoycho Draganoff

Date

September 14, 2005

## CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class:  
mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on this September 14, 2005  
date:

Type or printed name

Teresa Austin

Signature

Teresa Austin

Date

September 14, 2005



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:

Confirmation No.: 1893

Jyotirmoy PAUL, et al.

Group Art Unit: 2176

Serial No.: 09/631,884

Examiner: Nathan HILLERY

Filed: August 4, 2000

Title: MULTI-DEVICE SUPPORT FOR MOBILE APPLICATIONS USING XML

**Mail Stop Appeal Brief – Patents**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

**APPEAL BRIEF**

Sir:

This Appeal Brief is submitted in support of the Notice of Appeal filed on July 14, 2005.

**I. REAL PARTY IN INTEREST**

Oracle International Corporation, of Redwood Shores, California, is the real party in interest.

**II. RELATED APPEALS AND INTERFERENCES**

A Notice of Appeal has been filed in a related application, Serial No. 09/872,066, filed on May 31, 2001 (Attorney Docket No. 50277-1557), which includes similar subject matter and is assigned to the Real Party of Interest of this Appeal.

An Appeal Brief has been filed in a related application, Serial No. 09/872,978, filed on May 31, 2001 (Attorney Docket No. 50277-1608), which includes similar subject matter and is assigned to the Real Party of Interest of this Appeal.

### **III. STATUS OF CLAIMS**

Claims 1-4, 6-10, 12-17, 19-20, and 23-28 are pending in the present application, were finally rejected in the Final Office Action mailed on April 14, 2005 and are the subject of this appeal.

### **IV. STATUS OF AMENDMENTS**

No amendments were filed after the Final Office Action.

### **V. SUMMARY OF CLAIMED SUBJECT MATTER**

The present application contains independent Claims 1, 8, 12, and 14.

#### **A. Independent Claims 1, 12 and 14**

Claims 1, 12 and 14 recite similar features, except in the context of a method, a system, and a computer-readable medium, respectively. Claims 1, 12 and 14 are directed generally to an approach for allowing multiple types of clients to use a database application without hard-coding presentation logic for each of the multiple types of clients into the database application.

Database applications are software applications that communicate with a database server to store data into and retrieve data from a database. Common database applications include accounting programs and inventory programs. Typically, database applications are designed to receive input from computer systems and fixed computer terminals. In order to

input the data into the database applications, a user often has to manually record the data, bring the manually recorded data to the place at which the fixed computer system is located, and enter the manually recorded data into the fixed computer terminal. (Specification, page 1, lines 13-19).

To make data entry more efficient, mobile devices may be used to input data into the database application. For example, a hand-held bar code reader could be used to scan bar codes from labels on merchandise in a warehouse to have inventory data input directly into an inventory program. However, for such hand-held devices to be used with a database application, the database application has to be designed to support the devices.

Unfortunately, it is cost prohibitive to try to duplicate all the functionality of a database application for each possible mobile platform. (Specification, page 1, line 20 to page 2, line 3).

There are several types of hand-held devices that are used in the industry which have different hardware and software capabilities. These several types of hand-held devices may be described through a reference to a set of hypothetical devices, which set comprises, in increasing level of support for customization, the following hypothetical devices:

- **DT** - these are dumb terminals that support a particular protocol, such as Telnet;
- **HT1** - these are terminals that support data entry and formatting, such as devices supporting HTML 3.0 (and above) or similar protocols;
- **HT2** - these are same as HT1 terminals but further provide scripting functionality, such as devices supporting HTML 3.0 (and above) and JavaScript or VBScript;
- **JT** - these are same as HT2 terminals but further provide built in JVM and support for running Java Applets;

- **GT** - these are general-purpose devices that provide their own software development kit (“SDK”) to write applications. (Specification, page 6, lines 1-19).

The inventions recited in Claims 1, 12 and 14 address the problem of how to allow mobile devices to use database applications without having to specifically design the database applications to support all forms of mobile devices. This problem is addressed by an approach in which the data coming out of a database is converted into eXtensible Markup Language (XML). The XML data is independent of the type of the device to which the data is to be provided. XSL style sheets are written for each type of device to which XML data is to be provided. Prior to providing the data to the device, the appropriate XSL style sheet is applied to the device-independent XML data to convert it into the format required by the device to which the data is provided. This approach clearly separates the application logic from the device-specific presentation logic. Supporting applications on any new device with its own protocols only requires writing the corresponding XSL style sheets for that new device, as opposed to rewriting the interface logic of the database application to provide specific User Interface (UI) support for the device. For example, according to one embodiment, generic display and control functions are generated using a scripting language that HT2, JT and GT terminals support, and these functions are used along with the XSL style sheets to provide a user interface and data in a manner that can be handled by the mobile device. Further, using scripting and maintaining state on the client reduces redundant information exchange between a client and the server (Specification, page 3, lines 1-8; page 9, line 15 to page 10, line 2).

**B. Independent Claim 8**

Claim 8 is directed generally to a method for using a database application by clients

that support multiple markup language interpreters without hard-coding into the database application logic to support each of the multiple markup language interpreters.

The invention recited in Claim 8 addresses a similar problem as the problem addressed by the inventions recited in Claims 1, 12, and 14. Specifically, the invention recited in Claim 8 addresses the problem of how to allow mobile devices that may support different markup language interpreters to use a database application without having to hard-code logic in the database application to support each of the different markup language interpreters. A mobile UI server implements an application logic to convert data retrieved from a database application into XML data. Different XSL style sheets can be applied on the resulting XML data in order to support different types of mobile devices that use different protocols and markup languages. For example, different XSL style sheets may be applied to the resulting XML data to cause the XML data to be formatted in the manner required by a web server that supports web clients, Telnet server that supports telnet clients, or some other type of server that supports other type of clients. In order to support the existing mobile application functionality provided by the mobile UI server on any new type of mobile device that uses a new protocol and markup language, a new XSL style sheet need only be provided. (Specification, page 12, lines 1-17; FIGs. 1 and 2).

## **VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

A. Claims 1-3, 6-10, 12-16, 19-20, and 23-28 have been rejected under 35 U.S.C. § 103(a) as allegedly unpatentable over U.S. Patent No. 6,012,098 issued to Bayeh et al. ("BAYEH") in view of U.S. Patent No. 6,589,291 issued to Boag et al. ("BOAG"), further in

view of U.S. Patent No. 6,480,860 issued to Monday ("MONDAY"), and further in view of U.S. Patent No. 6,480,860 issued to Hill et al. ("HILL").

B. Claims 8-10 have been rejected under 35 U.S.C. § 103(a) as allegedly unpatentable over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL.

## VII. ARGUMENTS

### A. Introduction

It is well founded that to establish a *prima facie* case of obviousness under 35 U.S.C. § 103(a), all the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). In addition, a suggestion or motivation to make the claimed combination must be found in the prior art, not in applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

With respect to the present application, it is respectfully submitted that there is no suggestion or motivation to combine BAYEH with BOAG, MONDAY, and HILL, and for this reason Claims 1-3, 6-10, 12-16, 19-20, and 23-28 are not obvious under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL. It is further submitted that Claims 8-10 include one or more features that are not taught or suggested by BAYEH, BOAG, MONDAY, and HILL, whether taken alone or in combination, and for this additional reason Claims 8-10 are not obvious under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL.

**B. Claims 1-3, 6-10, 12-16, 19-20, and 23-28 Are Patentable Under 35 U.S.C. § 103(a) Over BAYEH In View of BOAG, Further In View of MONDAY, and Further In View of HILL**

It is respectfully submitted that Claims 1-3, 6-10, 12-16, 19-20, and 23-28 are patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL because there is no suggestion or motivation to combine BAYEH with BOAG, MONDAY, and HILL. Specifically, any combination of BAYEH with BOAG would change the principle of operation of BAYEH, and would render BAYEH unsatisfactory for its intended purpose.

1. There is no suggestion or motivation to combine BAYEH with BOAG, MONDAY, and HILL because the combination of BAYEH with BOAG would change the principle of operation of BAYEH

If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. *In re Ratti*, 270 F.2d 810, 813, 123 USPQ 349, 352 (CCPA 1959); MPEP §2143.01.

The Final Office Action asserts that it uses BAYEH as the primary reference in the rejections of Claims 1, 8, 12, and 14. In page 4, the Office Action further states that

It would have been obvious to one with ordinary skill in the art at the time of the invention to combine the invention of Bayeh et al. with that of Boag et al. because such a combination would allow dynamic determination of the most appropriate location for applying style sheets (first sentence of Boag et al.'s Abstract) used by the rendering servlet for parsing the XML data stream (last sentence of Bayeh et al.'s Abstract).



However, such a combination of BAYEH with BOAG would change the principle of operation of BAYEH.

BOAG provides “a technique for dynamically determining the most appropriate location to apply style sheets, whether that location is the client device, the server, or some combination thereof.” (BOAG, col. 4, lines 11-14.) BOAG further states in col. 4, lines 30-36, that

The technique comprises: selecting one or more style sheets to transform a particular input document; **determining whether a client device is capable of applying the selected style sheets; applying the selected style sheets at the client device when the determining has a positive result;** and applying the selected style sheets at a server when the determining has a negative result. (Emphasis added.)

Thus, the technique described in BOAG requires that: (1) a determination be always made whether to apply a style sheet to an input document at the client device or at the server, and (2) if a determination is made that the client device is capable of applying the style sheet, then the client device, and NOT the server, applies the style sheet to the input document.

On the other hand, in its FIG. 4, BAYEH depicts a system in which data servlet 83 retrieves data from a database, and formats the data into an XML data stream 97. The XML data stream 97 is then passed to rendering servlet 85. Rendering servlet 85 uses an XSL style sheet 99 to format the XML data stream 97 into an HTML data stream 96, which is sent to a browser 76 at the client 78. (See also BAYEH, col. 8, line 3 to col. 9, line 24).

Significantly, in col. 11, lines 35-43 BAYEH expressly states that:

According to the preferred embodiment, ***the rendering servlet must parse the XML data stream, and reformat it into HTML.*** This is ***necessary*** because browsers, by convention, expect to receive data that has been formatted with HTML. ***As discussed previously, this parsing process requires two types of data input: the XML data stream, and style sheet information.*** (Emphasis added.)

Thus, the technique described in BAYEH **expressly** teaches that the rendering servlet **NECESSARILY MUST** parse the XML data stream, which is received from the data retrieval servlet. Further, the above passage (in reference to the passage in col. 8, line 66 to col. 9, line 9) also **expressly** states that the parsing process **REQUIRES** two types of data input: the XML data stream **AND** style sheet information.

Thus, the system in BAYEH operates on the principle that the rendering servlet, which is different from the client browser (see BAYEH, FIG. 4), **NECESSARILY MUST** apply the style sheets to the XML data stream.

Suppose now that the technique of BOAG is applied to the system in BAYEH. According to BOAG's technique, the rendering servlet of BAYEH would **NOT** perform any rendering or formatting of the XML data stream by applying a style sheet to it when the client browser is capable of applying the style sheets. In this case, the rendering servlet would have to pass the XML data stream to the client browser without any formatting. This, however, clearly violates and changes the principle of operation of BAYEH, which requires that the rendering servlet **MUST** parse the XML data stream and reformat it into an HTML data stream by applying style sheet information.

Furthermore, BAYEH expressly states that this principle of operation, according to which the rendering servlet necessarily must format the incoming XML data stream into an HTML data stream by applying style sheets, is what brings about the advantages of BAYEH's invention. Specifically, in col. 9, lines 18-24, BAYEH states that

Because the [HTML] data stream 96' received by the [client] browser 76' uses the same formatting instructions supported by existing browser implementations, the processing model defined for the present invention *minimizes the extent of disruption to existing software by localizing all*

***changes to code running on servers. This minimized disruption further maximizes the advantages of the preferred embodiment.*** (Emphasis added.)

Thus, BAYEH expressly and unambiguously states that the advantages of BAYEH's system are directly dependant on the requirement that the rendering servlet must convert the XML data stream into an HTML stream by applying a style sheet. However, when the rendering servlet is changed so that it does not format the XML data stream as per the technique of BOAG, the resulting system would not provide the express advantages touted by BAYEH. For this reason, one of ordinary skill in the art would not consider it obvious to combine the technique of BOAG with the system in BAYEH because the resulting system would not provide the advantages expressly stated by BAYEH.

For the above reasons, it is respectfully submitted that there is no suggestion or motivation to combine BAYEH with BOAG because such combination would change the principle of operation of BAYEH.

2. There is no suggestion or motivation to combine BAYEH with BOAG, MONDAY, and HILL because the combination of BAYEH with BOAG would render BAYEH unsatisfactory for its intended purpose

If the proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984); MPEP §2143.01.

As discussed above, BAYEH expressly requires that the XML data stream produced by the data retrieval servlet MUST be converted into an HTML data stream before being sent to the client browser. BAYEH also expressly states that the HTML data stream is created

based on two inputs: the XML data stream received from the data rendering servlet, and an XSL style sheet. (Col. 8, line 66 to col. 9, line 9). Further, BAYEH expressly shows in FIG. 4 that the XSL style sheets are stored in a database that is accessible by the rendering servlet but NOT by the client browser.

As also discussed above, the technique in BOAG requires that a determination be always made whether to apply a style sheet to an input document at the client device or at the server, and that if the client device is capable of applying the style sheets, then the client device, and NOT the server, applies the style sheet to the input document.

Suppose, however, that the technique of BOAG is combined with BAYEH's system. According to BOAG's technique, when the client device is capable of applying a style sheet, then the client device, and not the rendering servlet, must apply the style sheet to the XML data stream of BAYEH. Thus, in the hypothetical BAYEH-BOAG system, the rendering servlet must pass an unformatted XML data stream to the client browser. However, the client browser does not have a style sheet with which to format the XML data stream and does not have access to the style sheets used by the rendering servlet. Further, the rendering servlet is incapable of passing any style sheets to the client browser because in BAYEH's system the rendering servlet sends to the client browser only an HTML data stream. Thus, the client browser would not know how to convert the XML data stream into an HTML document, and the end result would be that the client browser would not be able to display the data properly. Thus, the combination of BOAG with BAYEH would not only render BAYEH unsatisfactory for its intended purpose, but may even render it inoperable if the client browser is incapable of displaying raw XML data.

For the reasons set forth above, it is respectfully submitted that there is no suggestion or motivation to combine the primary reference BAYEH with BOAG because such combination would violate the principle of operation of BAYEH and would render BAYEH unsatisfactory for its intended purpose.

For at least the foregoing reasons, Claims 1-3, 6-10, 12-16, 19-20, and 23-28 are patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL.

Further, the Final Office Action has rejected Claims 4 and 17 as allegedly unpatentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, further in view of HILL, and further in view of Karanjit Siyan, *NetWare TCP/IP and NFS*, New Riders Publishing 1994, pp. 11, 94, 103 ("SIYAN"). However, since as shown above there is no suggestion or motivation to combine BAYEH with BOAG, Claims 4 and 17 are also patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, further in view of HILL, and further in view of SIYAN.

Therefore, all of Claims 1-4, 6-10, 12-17, 19-20, and 23-28 are patentable over the references of record.

**C. Claims 8-10 Are Patentable Under 35 U.S.C. § 103(a) Over BAYEH In View of BOAG, Further In View of MONDAY, and Further In View of HILL**

It is respectfully submitted that Claims 8-10 are patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL because BAYEH, BOAG, MONDAY, and HILL, whether taken alone or in combination do not teach or suggest all features included in these claims.

1. Independent Claim 8

Claim 8 includes the features of:

...  
wherein **a plurality of mark-up languages are each associated with one or more client device types** of a plurality of client device types;

...  
**selecting**, based on a client device type to which the output is to be sent, **a second mark-up language of said plurality of mark-up languages** that is different than said first mark-up language;

....

It is respectfully submitted that none of BAYEH, BOAG, MONDAY, or HILL describes the features of (1) a plurality of mark-up languages that are each associated with one or more client device types, and (2) selecting, based on a client device type to which the output is to be sent, a second mark-up language of said plurality of mark-up languages.

In page 9, numbered paragraph 10, the Final Office Action states that the rejection of Claim 8 is based on the rationale of the rejection of Claim 1. Thus, the Office Action does not specify exactly what in BAYEH, BOAG, MONDAY, or HILL corresponds to the feature of Claim 8 of a plurality of mark-up languages that are each associated with one or more client device types.

It seems that the Final Office Action analogizes the style sheets, which are used in HILL to adapt the content of a document to a particular monitor or display, to the plurality of markup languages recited in Claim 8. It is respectfully submitted that this analogy is incorrect.

First, the Applicants respectfully submit that style sheets and markup languages are two distinct and separate concepts that are not equivalent. For example, in col. 2, lines 22-28, BOAG states:

Style sheet languages such as XSL, along with their associated processors, are powerful tools for filtering data content encoded in notations such as XML, as well as for transforming documents encoded into other markup languages such as HTML (HyperText Markup Language) or WML (Wireless Markup Language).

Thus, while the above passage may be showing that style sheets may be used for transforming documents from one markup language to another, the above passage also makes it clear that style sheets are not equivalent to markup languages.

Second, contrary to the implicit assertion in the Office Action, HILL does not describe that a plurality of markup languages are each associated with one or more client device types. In the passage in col. 11, lines 4-23, which the Office Action asserts as showing a plurality of markup languages, HILL states that

The document, the layout generator and the style sheets may be created by the author. The author may create a layout generator which selects **a different style sheet for each type of display**. Alternatively, the author may create a layout generator which selects **the same style sheet for all display devices** with capabilities within a predetermined range. For example, **the author may determine that a style sheet entitled "High Resolution" may be used for all display devices with resolutions within a first predetermined range, a style sheet entitled "Medium Resolution" may be used for all display devices with resolutions within a second predetermined range, and a style sheet entitled "Low Resolution" may be used for all other display devices**. An authoring tool may assist the author in creating the layout generator and the style sheets. The layout generator may be designed to work with a particular document and a particular set of style sheets or style definitions. Alternatively, the layout generator may be a general purpose layout generator which is designed to work with multiple documents and different sets of style sheets or style definitions. (Emphasis added.)

Thus, while the above passage may be showing that a different style sheet may associated with a monitor or display of a different resolution, there is nothing in this passage to suggest that a different **markup language** is associated with each of the displays based on the display resolution. On the contrary, throughout HILL, the only markup language discussed is

HTML as it is being used by browsers, such as browser 206 depicted in FIGs. 2 and 3 of HILL. Thus, there is nothing in HILL to suggest that a **mark-up language** may be associated with the type of the display or monitor that is used by the client.

In contrast, Claim 8 recites the feature of **a plurality of mark-up languages that are each associated with one or more client device types** of a plurality of client device types. Furthermore, since HILL does not show the feature of a plurality of mark-up languages, HILL cannot possibly show the feature of Claim 8 of **selecting**, based on a client device type to which the output is to be sent, **a second mark-up language of said plurality of mark-up languages** that is different than said first mark-up language.

For these reasons, the Applicants respectfully submit that BAYEH, BOAG, MONDAY, and HILL, whether taken separately or in combination, fail to describe all features of Claim 8. Thus, Claim 8 is patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL.

2. Dependent Claims 9-10

Claims 9-10 depend from independent Claim 8, and thus include all of the features of Claim 8. It is therefore respectfully submitted that Claims 9-10 are patentable under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL for at least the reasons set forth herein with respect to Claim 8.

**VIII. CONCLUSION AND PRAYER FOR RELIEF**

Based on the foregoing, it is respectfully submitted that the rejections of Claims 1-4, 6-10, 12-17, 19-20, and 23-28 under 35 U.S.C. § 103(a) as being unpatentable over the art of record lack the requisite factual and legal bases.



Appellants therefore respectfully request that the Honorable Board reverse the rejection of Claims 1-3, 6-10, 12-16, 19-20, and 23-28 under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, and further in view of HILL.

Appellants also respectfully request that the Honorable Board reverse the rejection of Claims 4 and 17 under 35 U.S.C. § 103(a) over BAYEH in view of BOAG, further in view of MONDAY, further in view of HILL, and further in view of SIYAN.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

Date: September 14, 2005

Stoycho D. Draganoff  
Stoycho D. Draganoff  
Reg. No. 56,181

2055 Gateway Place, Suite 550  
San Jose, California 95110-1089  
Tel: (408) 414-1080 ext. 208  
Fax: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: **Mail Stop Appeal Brief – Patents**, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

on Sept 14, 2005 by Jessie Huest

**CLAIMS APPENDIX**

1     1.     A method for allowing multiple types of clients to use a database application without  
2           hard-coding presentation logic for each of the multiple types of clients into the  
3           database application, the method comprising the steps of:  
4           prior to providing data from the database application to a particular client, performing  
5           the steps of:  
6                 converting the data that is to be transmitted from the database application to  
7                         the particular client into an XML output without regard to the device  
8                         type of the particular client by:  
9                                 identifying a data type to which the data corresponds, wherein  
10   the data type reflects a type of the data that is read out  
11   of the database;  
12                                 selecting from a plurality of document type definitions, a  
13   document type definition associated with said data type;  
14                                 and  
15                                 converting the data to XML output based on said selected  
16                                 document type definition;  
17                 identifying the particular client device type of the particular client;  
18                 wherein sets of metadata are each associated with a client device type of a  
19                         plurality of client device types and indicates how to convert said XML  
20                         output to output for the client device type;  
21                 selecting, based on the particular client device type, a particular set of  
22                         metadata from among the sets of metadata;  
23                 reading the particular set of metadata; and  
24                 based on the particular set of metadata, converting the XML output to output  
25                         for the particular client device type; and  
26                 providing the output for the particular client device type to the particular client.

1     2.     The method of Claim 1 wherein:

50277-0352 (OID-1999-129-01)

2 the step of reading the particular set of metadata includes reading an XSL style sheet  
3 associated with said particular client device type; and  
4 the step of converting the XML output includes applying the XSL style sheet to said  
5 XML output.

1 3. The method of Claim 1 wherein the step of converting the data that is to be  
2 transmitted from the database application to the particular client into an XML output  
3 includes converting the data based on one or more document type definition files.

1 4. The method of Claim 1 wherein:  
2 the particular client is a Telnet client;  
3 the Telnet client communicates with a Telnet server to request data from said database  
4 application; and  
5 the step of providing said output to said particular client includes the steps of  
6 sending the output to said Telnet server using a recipient specific format; and  
7 said Telnet server providing said output to said Telnet client.

1 5. (Canceled)

1 6. The method of Claim 1 wherein the XML output includes display instruction data  
2 indicating that said data is to be displayed in a first manner.

1 7. The method of Claim 6 wherein the step of converting the XML output includes the  
2 step of generating output for said particular client device type that causes said data to  
3 be displayed in a second manner that is different than said first manner when said  
4 particular client device type is not able to display said data in the first manner.

1 8. A method for using a database application with clients that support multiple mark-up  
2 language interpreters without hard-coding into the database application logic to  
3 support each of the multiple mark-up language interpreters, the method comprising

4 the steps of:  
5 converting output of the database application to first data that conforms to a first  
6 mark-up language without regard to the type of mark-up language interpreter  
7 supported by a client to which the output is to be sent by:  
8 identifying a data type to which the data corresponds, wherein the data  
9 type reflects a type of the data that is read out of the database;  
10 selecting from a plurality of document type definitions, a document  
11 type definition associated with said data type; and  
12 converting the data to XML output based on said selected document  
13 type definition;  
14 wherein a plurality of mark-up languages are each associated with one or more client  
15 device types of a plurality of client device types;  
16 selecting, based on a client device type to which the output is to be sent, a second  
17 mark-up language of said plurality of mark-up languages that is different than  
18 said first mark-up language;  
19 converting the first data to second data that conforms to the second mark-up language;  
20 and  
21 sending the second data to the client.

1 9. The method of Claim 8 wherein the step of converting the first data to second data is  
2 performed by applying an XSL style sheet to said first data.

1 10. The method of Claim 8 wherein the step of sending the second data to the client  
2 includes sending the data to a server to which the client is connected through a  
3 wireless connection, and then sending the data from the server to the client over said  
4 wireless connection.

1 11. (Canceled)

1 12. A system comprising:

2 a database system;  
3 a database application operatively coupled to said database system;  
4 said database application including:  
5 application logic that:  
6 retrieves data from said database system to produce a first output in a  
7 format that is independent of a type of client device that is to  
8 receive the output;  
9 an XML processor that:  
10 identifies a data type to which the data retrieved from the database  
11 corresponds;  
12 identifies a document type definition associated with said data type;  
13 and  
14 applies the document type definition to the data from the database,  
15 thereby formatting the first output into XML to produce second  
16 output that is independent of the type of client device is to  
17 receive the output; and  
18 an XSL processor that converts the second output into a third output based on  
19 an XSL style sheet associated with the type of client device that is to  
20 receive the output; wherein the XSL style sheet is selected based on the  
21 type of client device.

- 1 13. The system of Claim 12 further comprising:  
2 a plurality of servers operatively coupled to said database application;  
3 said plurality of servers including at least a first server configured to provide services  
4 to clients that support a first protocol and a second server configured to  
5 provide services to clients that support a second protocol that is different from  
6 said first protocol; and  
7 a plurality of clients including a first client that interacts with said database  
8 application through said first server and a second client that interacts with said  
9 database application through said second server.

1 14. A computer-readable medium carrying instructions for allowing multiple types of  
2 clients to use a database application without hard-coding presentation logic for each  
3 of the multiple types of clients into the database application, the instructions including  
4 instructions for performing the steps of:  
5 prior to providing data from the database application to a particular client, performing  
6 the steps of:  
7 converting the data that is to be transmitted from the database application to  
8 the particular client into an XML output without regard to the device  
9 type of the particular client by:  
10 identifying a data type to which the data corresponds;  
11 selecting from a plurality of document type definitions, a  
12 document type definition associated with said data type;  
13 and  
14 converting the data to XML output based on said selected  
15 document type definition;  
16 identifying the particular client device type of the particular client;  
17 wherein sets of metadata are each associated with a client device type of a  
18 plurality of client device types and indicates how to convert said XML  
19 output to output for the client device type;  
20 selecting, based on the particular client device type, a particular set of  
21 metadata from among the sets of metadata;  
22 reading the particular set of metadata; and  
23 based on the particular set of metadata, converting the XML output to output  
24 for the particular client device type; and  
25 providing the output for the particular client device type to the particular client.

1 15. The computer-readable medium of Claim 14 wherein:  
2 the step of reading the particular set of metadata includes reading an XSL style sheet  
3 associated with said particular client device type; and

4 the step of converting the XML output includes applying the XSL style sheet to said  
5 XML output.

1 16. The computer-readable medium of Claim 14 wherein the step of converting the data  
2 that is to be transmitted from the database application to the particular client into an  
3 XML output includes converting the data based on one or more document type  
4 definition files.

1 17. The computer-readable medium of Claim 14 wherein:  
2 the particular client is a Telnet client;  
3 the Telnet client communicates with a Telnet server to request data from said database  
4 application; and  
5 the step of providing said output to said particular client includes the steps of  
6 sending the output to said Telnet server using a recipient specific format; and  
7 said Telnet server providing said output to said Telnet client.

1 18. (Canceled)

1 19. The computer-readable medium of Claim 14 wherein the XML output includes  
2 display instruction data indicating that said data is to be displayed in a first manner.

1 20. The computer-readable medium of Claim 19 wherein the step of converting the XML  
2 output includes the step of generating output for said particular\_client device type that  
3 causes said data to be displayed in a second manner that is different than said first  
4 manner when said particular client device type is not able to display said data in the  
5 first manner.

1 21. (Canceled)

1 22. (Canceled)

- 1 23. The method of Claim 1, wherein the particular client device type indicates at least one  
2 of a dumb terminal, a telnet terminal, a bar code scanner and a browser-less device.
- 1 24. The system of Claim 12, wherein the type of client comprises at least one of a dumb  
2 terminal, a telnet terminal, a bar code scanner and a browser-less device.
- 1 25. The computer readable medium of Claim 14, wherein the particular client device type  
2 indicates at least one of a dumb terminal, a telnet terminal, a bar code scanner and a  
3 browser-less device.
- 1 26. The method of Claim 1, wherein the data type indicates at least one of a data entry  
2 form, a queried data, a message, a form level query data and a field level query data.
- 1 27. The system of Claim 12, wherein the data type indicates at least one of a data entry  
2 form, a queried data, a message, a form level query data and a field level query data.
- 1 28. The computer readable medium of Claim 14, wherein the data type indicates at least  
2 one of a data entry form, a queried data, a message, a form level query data and a  
3 field level query data.